Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 83 (2016) 705 – 711

Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

# A lightweight security protocol for NFC-based mobile payments

Mohamad Badra[a], Rouba Borghol Badra[b,]*

[a]*Zayed University, Dubai, UAE*
[a]*RIT, Dubai, UAE*

**Abstract**

In this work, we describe a security solution that can be used to securely establish mobile payment transactions over the Near-Field Communication (NFC) radio interface. The proposed solution is very lightweight one; it uses symmetric cryptographic primitives on devices having memory and CPU resources limitations. We show that our approach maintains the security of NFC communications and we further demonstrate that our solution is simple, scalable, cost-effective, and incurs minimal computational processing overheads.


Peer-review under responsibility of the Conference Program Chairs

*Keywords:* Near-Field Communication (NFC); Key Exchange; Authentication; Eavesdropping; Relay Attack; Transport Layer Security (TLS).

## 1. Introduction

Over the last decade, we have witnessed a rapid emergence of mobile/wireless access and applications/services that have fueled the explosive growth in the number of mobile's users. Wireless communication technologies are paving the way for the development of innovative, interactive and smarter applications and architectures. Nevertheless, many of the emerging wireless services are prone to unauthorized access and eavesdropping are easier as compared to wired communication technologies because a) wireless data is transmitted over the air and usually there is no physical controls over the boundaries of transmissions[1], b) security features designed for wireless communications are sometimes poor, and c) attackers don't have to tap into the network (i.e., due the broadcast nature of radio propagation) to insert rogue wireless access points, increasing the potential for unauthorized access to the transmission[2].

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
*E-mail address:* mbadra@gmail.com


Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.156

In this paper, we study the security of mobile payment systems that are based on short-range wireless communication technologies. In particular, we focus on Near-Field Communication (NFC)[3] that enables contactless transactions and messages' exchange between two devices at the range of few centimeters. In the NFC communication architecture, a range of devices such as smartphones can implement the NFC standard. The devices can be in active or passive modes. When in active mode, both devices can generate a Radio Frequency (RF) field and start data exchange (role of Initiator), and when in passive mode (e.g., tag or contactless card), only the active device can initiate communication sessions and the role of the passive device is called a target (e.g., ticket counter, code bar, etc.). NFC devices can operate in three modes: reader/writer (an active device communicates with a passive device), peer-to-peer (active devices communicate with passive or active devices) and optional card emulation modes (device emulates a passive element, such as a contactless card[4]).

NFC-enabled mobile devices allow a wide range of applications including, but not limited to, mobile contactless payment, mobile ticketing, information exchange, and access control. NFC-enabled devices and contactless Point-of-Sales (PoS) terminals execute payment transactions using communication protocols that are based on Radio-Frequency IDentification (RFID) standards such as ISO/IEC 14443[3].

Figure 1 shows the case of a mobile device that integrates with the NFC technology. This device is typically composed of various integrated circuits, an NFC interface and Secure Elements (SE). The communications between NFC devices are enabled over the NFC interface. This interface is composed of an NFC Contactless Front-end, an NFC antenna and an NFC controller. NFC-enabled devices incorporate SE to a) securely store confidential information such as user account information, and b) to connect to the NFC controller to perform secure proximity transactions with external NFC devices. The Single Wire Protocol (SWP) [5] is used as an interface between the SE and the NFC controller enabling the communications between the payment application installed on the SE and the contactless readers (e.g., POS) through the NFC interface. The host controller is the part of the NFC system that processes exchanged data and establishes connections between the NFC controller and the SE. The NFC controller is connected to the host controller through the Host Controller Interface (HCI), which is also used for the communication between the NFC controller and SEs over the SWP interface. The UICC uses the ISO 7816 Interface to exchange data to a remote server in the Network. This interface includes a set of Data Packet called Application Protocol Data Unit (APDU) [6] for reading, writing and exchanging data between the host and the UICC card.

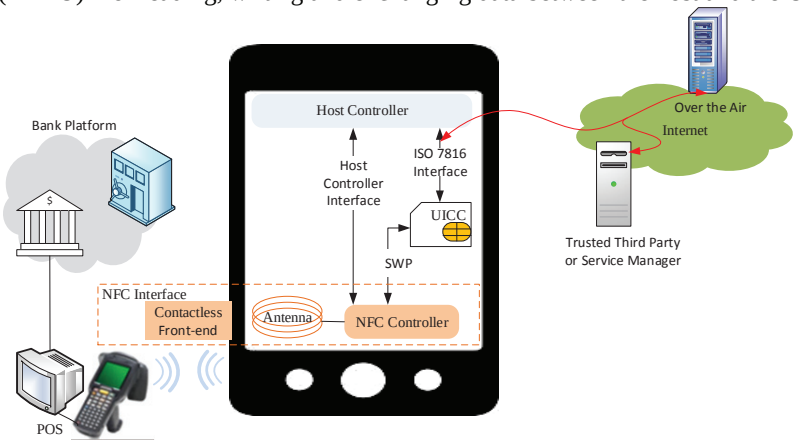


Fig. 1. Architecture of NFC-enabled mobile phone with UICC as a Secure Element.

The lower layers of NFC does not support link-level security primitives[7] which allows unscrupulous attackers to exploit vulnerabilities of eavesdropping and intercepting the data as discussed later or even having the device taken over completely by the attackers. Moreover, there are also concerns related to the privacy of the users because third parties are able to determine the current activities of the consumer and learn about user's behavior.

The rest of this paper is organized as follows. In Section 2 we discuss some approaches that have been recently proposed to protect the NFC-based mobile payments. In Section 3, we review the security issues of the NFC technology and we show its vulnerability to multiple attacks. Section 4 describe the design of our proposed solution

to secure NFC communications. In Section 5 we discuss the security and performance of our solution. Finally, our concluding remarks are presented in Section 6.

## 2. Related works

NFC mobile payments involve collaborations between mobile phone manufacturers and OS vendors (e.g., Google), mobile phone operators, and banking organizations (e.g., VISA)[3]. When used for mobile contactless payment, NFC-enabled devices incorporate SEs to securely store private information. Using NFC in the form of a credit card means that someone could forgo credit cards altogether and instead make payments using mobile devices or smartphones[8]. In the case of mobile payments, the user holds or taps the phone close to the merchant's reader[8] to start sending his or her account information to the contactless PoS reader through RF link. The traditional payment and settlement processes are the same used by the mobile payment applications.

Authors of[9] indicated that all payment applications are protected from a network adversary via the use of Transport Layer Security (TLS)[10]. In[11], the authors proposed the use of Elliptic Curve-based signature to sign exchanged NFC data and therefore to protect it against several attacks such as manipulation and to provide integrity and authenticity to NFC exchanged messages. In[12], the authors propose a tiny TLS stack embedded in SE, which securely runs the mobile payment application. In[13], the authors describes a certificate-based solution for enabling the client to send signed messages. The authors introduce the use of X.509 short lived certificates on the client side to eliminate the cost related to the certificate validation and to the communication and computational operations.

The aforementioned solutions use a certificate-based authentication method to authenticate the SE and the PoS. However, using certificates impose an undesirable constraints on SE embedded in the mobile devices, which often have memory and battery-life limitations. Moreover, some other major limitations will influence the deployment and performance of NFC communications such as the processing time required to achieve the public-key cryptographic operations. More discussions on the issues of using certificates in wireless networks can be found in[14].

### *2.1. Contributions to this work*

Despite the fact that NFC is a short-range radio technology, NFC is still susceptible to several threats and vulnerabilities such as eavesdropping and man-in-the-middle attacks. In this paper, we a) identify the threats and vulnerabilities of the air interface between NFC-enabled devices and the PoS, b) propose a solution to secure NFC-based payment transactions with Smartphones and c) compare our solution to the solution presented in[12].

## 3. NFC Security Issues: Vulnerabilities and Attacks

Despite the various applications that can be operated using NFC technology, the fact that the lower layers of NFC include no communication security primitives makes this technology susceptible to a wide range of vulnerabilities and attacks[18]. In this paper, the customer and the devices are assumed to be trusted; we focus on the security concerns related to the transactions over the NFC radio interface (Fig. 1). We note that different other vectors are possible for attacking the NFC. This mainly includes 1) malicious application installed on the NFC-enabled devices instead of the legitimate application such as malware applications[9], 2) side channels over shared hardware components such as smart cards to extract or overwrite the secret and financial information stored into the cards, and 3) malicious Operating Systems where the attacker can gain privileged access to the device and then exploit vulnerabilities[15]. In[16], the authors discuss some threats for smartphones such as data leakage, unintentional disclosure of data, attacks on decommissioned devices, surveillance attacks and financial malware attacks. Existing solutions[16, 17] can be used to mitigate the aforementioned issues.

The NFC tags could contain malicious threats in a way similar to Internet browsers and malicious URLs. For example, one can spoof the content of NFC tags or replace the original tags to redirect users to the attacker website, initiate phone calls, send Short Messages (SMS), and surreptitiously install malicious code (e.g., Worms and Viruses) on the NFC-enabled device without requiring user consent.

An attacker can passively eavesdrop on the communication between the NFC-enabled devices without any manipulation or modification to the data being exchanged between the devices (Fig. 2). Since NFC data is conveyed

through radio transmission, and by itself, cannot be protected against eavesdropping, attackers can use an antenna to sniff data exchanged between NFC-enabled devices.

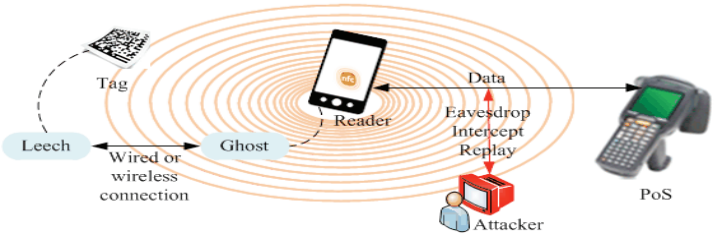


Fig. 2. Attack vulnerabilities on NFC payment systems.

Instead of being passive, an attacker can act as an active eavesdropper that is able to intercept and modify the data being exchanged between NFC devices and to inject new data (Fig. 2). Replay attacks are similar to a passive man-in-the-middle attach where the attacker records a valid NFC signal for later use. Relay attack could be installed to establish a bi-directional communication channel between two legitimate NFC-enabled devices in order to create a relay between two NFC-enabled devices or between a reader and a tag.

The most efficient solution to make it more resilient to most of the above security issues is by the establishment of secure sessions over the NFC radio interface.

## 4. Our Proposed Security Solution

Cryptography is the main mechanism that should be used to protect sensitive payment applications and users' account data. In the case of mobile payment ecosystem, the physical credit card is replaced with another element, in which the NFC-enabled mobile device emulates the card and stores the user's data into a Secure Element. Data stored into the SE could be managed by third parties and multiple contactless applications can be stored and executed on the secure element.

In[14], we have introduced an extension to the TLS protocol in order to establish a secure session between the SE (e.g., SIM card) and the OTA server, in which case the mobile device is acting as a passive proxy. In the following sections, we describe our proposed solution which aims to prevent the security attacks that are related to the NFC interface. Our solution makes use of certificate-based authentication between a PoS and a trusted third party (TTP) (e.g., trusted service manager) and of shared-secret-based authentication between the third party and the NFC-enabled device (Fig. 3). We assume that the secret key shared between the TTP and the mobile (i. e., $SK_{TTP\_SE}$) is securely stored into SE and that cryptographic computations are performed inside SE. We also assume that SE offers good tamper-resistant and that certain physical hardware and software protections are used to make it difficult to extract or modify private and secret information in SE. We assume either the mobile device or the PoS or both have Internet connection. When the user connects to PoS to perform a mobile payment, the PoS may or may not have an Internet connection. We explain our proposed solution in the case where the PoS has an Internet connection and in the case where it hasn't an Internet connection.

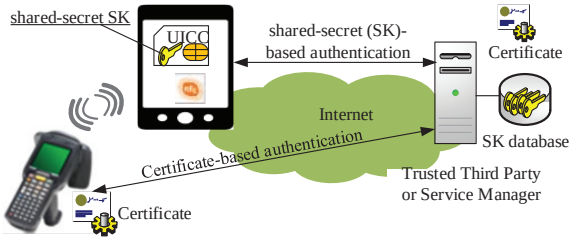


Fig. 3. Shared-secret and certificate-based authentication in our proposed security solution.

### 4.1. Case 1: Our authentication solution when PoS has an Internet connection

To mutually authenticate, the TTP, SE, and the PoS engage in our protocol as described below (Fig. 4).

- Step 1: The SE sends the identity of the user and a random value to the PoS (i.e., $RV_{SE}$, $ID_{SE}$).
- Step 2: The PoS generates a random value ($RV_{PoS}$) and sends it to the TTP along with its certificate and the random value and the identifier received from SE (i.e., $Cert_{PoS}$, $RV_{PoS}$, $RV_{SE}$, $ID_{SE}$).
- Step 3: The TTP verifies the PoS certificate, generates a secret key $SK_{PoS\_SE}$. Then, the TTP computes a session_key by applying a PRF (Pseudo Random Function) on $RV_{PoS}$, $RV_{SE}$, $ID_{SE}$ and the XORing of $SK_{TTP\_SE}$ and $SK_{PoS\_SE}$. Next, the TTP encrypts the concatenation of session_key and $SK_{PoS\_SE}$ using the PoS public key ($Pub_{PoS}$) and sends the encrypted result to the PoS (i.e., *E*).
- Step 4: Upon receipt of E, the PoS decrypts it using its private key ($Pr_{PoS}$) by computing AsymD(E, $Pr_{PoS}$). Then, the PoS symmetrically encrypts the session_key using $SK_{PoS\_SE}$ and sends the result (i.e., *F*) to SE.
- Step 5: SE computes the session_key and then symmetrically decrypts *F* to obtain $SK_{PoS\_SE}$, which is used therefore to securely exchange data between the SE and the PoS.

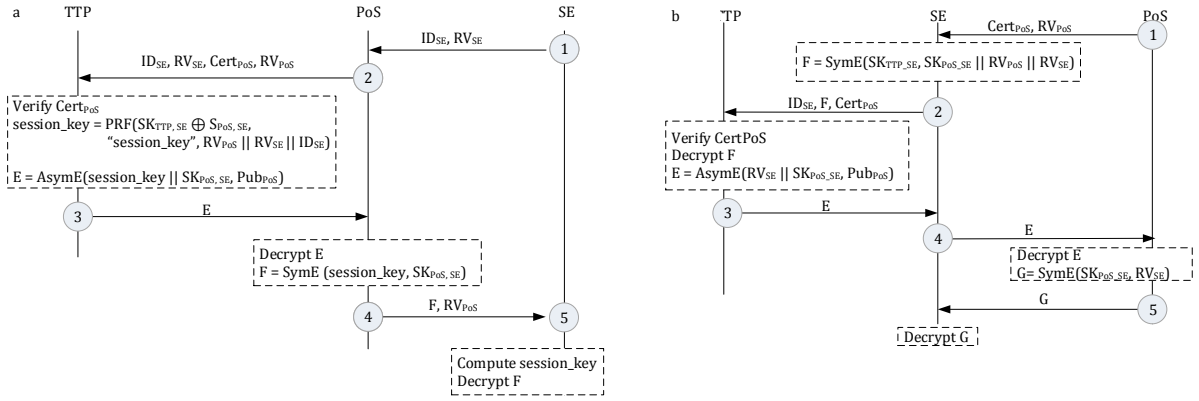


Fig. 4. (a) Case 1; (b) Case 2.

### 4.2. Case 2: Our authentication solution when PoS has no Internet connection

To mutually authenticate, the TTP, SE, and the PoS engage in our protocol as described below (Fig. 4).

- Step 1: The PoS sends its certificate and a random value to the SE ($Cert_{PoS}$, $RV_{PoS}$).
- Step 2: The SE generates a random value $RV_{SE}$ and a secret key $SK_{PoS\_SE}$ and symmetrically encrypts the concatenation of $RV_{SE, RVPoS}$, and $SK_{PoS\_SE}$, using $SK_{TTP\_SE}$. SE sends the encrypted value *F* along with its $ID_{SE}$ and $Cert_{PoS}$ to TTP.
- Step 3: TTP verifies the PoS certificate and then symmetrically decrypts *F* using $SK_{SE\_TTP}$. Then, the TTP encrypts the concatenation of $SK_{PoS\_SE}$ and $RV_{SE}$ using the PoS public key ($Pub_{PoS}$) and sends *E* to SE.
- Step 4: The SE forwards *E* to PoS.
- Step 5: Upon receipt of *E*, the PoS asymmetrically decrypts it using its private key ($Pr_{PoS}$) by computing AsymD(E, $Pr_{PoS}$). Next, PoS symmetrically encrypts $RV_{SE}$ using $SK_{PoS\_SE}$ and sends the encrypted value (i.e., *G*) to SE which symmetrically decrypts it and compares it for equality to the random value initially generated. If the two random values are equal, then the PoS is authenticated.

Once the authentication is successful, both the SE and PoS compute an encryption key that is derived from $SK_{PoS\_SE}$ in order to encrypt subsequent session traffic.

## 5. Evaluation and Analysis of our Solution

In this section, we evaluate our proposed approach, in particular we demonstrate that it is resilient against the attacks previously discusses such as replay attack, man-in-the-middle attack, and known session-key attack. We later evaluate its performance and compare it with other proposed approaches.

An attacker can impersonate a malicious PoS to a victim SE. Our solution is resilient against man-in-the-middle attacks described earlier since on the first hand, it is fairly very difficult for man-the-middle-attacks to decrypt the session_key being asymmetrically encrypted. On the second hand, our protocol provides a mutual authentication between SE and PoS and the TTP as well. The mutual authentication between SE and PoS is established by applying the PRF function associated with a key on data being exchanged between both the two entities. Hence, it is not be possible for man-in-the-middle attacks to falsify the exchanged messages without being detected.

An attacker may record the data being exchanged between the different parties and save them for later use. However, it is useless for the attacker to use the recorded data, especially when using sequence numbers or random values that include timestamps to generate a fresh encryption key, in which the attacker needs to know the secret key to compute the same encryption key.

As previously discussed, NFC by itself cannot protect against eavesdropping. However, eavesdropping can be avoided by encrypting exchanged data with an encryption key that is derived from the session_key as previously described.

### *5.1. Performance Evaluation of our Solution*

We evaluate the performance of our solution and compare it with the proposed approaches in[12]. Our solution introduces the use of:

- Symmetric encryption to mutually authentication the SE and the TTP.
- Certificates to authenticate the PoS to the TTP.

Our comparison focuses on evaluating the computational costs associated with encryption and decryption operations. The costs (on per device basis) are compared with[12] in Table II ($t_e/t_d$, $t_s$ / $t_v$, and $t_{ae}$ / $t_{ad}$ are respectively the symmetric encryption / decryption, signature / verification computing time, and asymmetric encryption / decryption computing time). It can be seen that our proposed scheme performs better by incurring lower computational costs. In fact, the asymmetric cryptographic operations are executed on either the PoS or the TTP, but not on the SE which has memory and CPU resources limitations.

In order to establish a mutual authentication, our solution requires the SE to execute two symmetric encryption operations, whereas the solution in[12] requires one certificate-based signature and verification, which are more expensive in term of computational cost. In general, symmetric encryption tends to be 1000 times[19] faster than asymmetric encryption.

Our solution can easily be extended to protect the identity protection from eavesdroppers. For this end, the SE and the TTP can implement an anonymous credential technique to remove any link between the current communication and the SE identity. In the case of [12], client identity protection is not possible without renegotiating a new TLS session. However, renegotiating a new TLS session requires more asymmetric cryptographic computations which are the rate limiting step in TLS, and therefore, the renegotiation has negative performance consequences. In fact, renegotiation requires another round of an asymmetric encryption/decryption, which means the double number of asymmetric en-/decryption operations for TLS Handshake message processing, for both server and client. Moreover, renegotiation requires twice the number of messages and roundtrips than a single TLS handshake, thus significantly increasing the overall delay in the session setup.

Table 2. Cryptographic Cost Comparison between our solution and[12].

| Entity | Our Solution case 1 | Our Solution case 2 | Solution in[12] |
|---|---|---|---|
| SE | $1t_e + 1t_{PRF}$ | $1t_e + 1t_d$ | $1t_{PRF} + 1t_{ae} + 1t_s$ |
| PoS | $1t_e + 1t_{ad}$ | $1t_e + 1t_{ad}$ | $1t_{PRF} + 1t_{ad} + 1t_v$ |
| TTP | $1t_v + 1t_{PRF} + 1t_{ae}$ | $1t_v + 1t_d + 1t_{ae}$ | NA |

## 6. Conclusion

NFC-based mobile payment systems have generated considerable interests in data security issues of NFC's users. The security issues are mostly related to the fact that NFC specifications specify no communication security primitives and consequently the technology is susceptible to a wide range of vulnerabilities and attacks. In this context, we presented a security solution that enhances the security of NFC transactions as well as the privacy of users. We demonstrate that our solution is more efficient than existing solutions in terms of reducing communication and computational costs. We argue that our proposed design is suitable for resource-constrained devices such as the Security Elements being embedded into the NFC-enabled devices. Our future works consist of integrating our solution into TLS protocol using pre-shared secrets[20, 21, 22].

## Acknowledgements

The authors would like to express his gratitude to Sheraly Zeadally for his valuable feedback and comments which helped to improve the quality and presentation of this paper.

## References

1. Karygiannis T and Owens L. Wireless Network Security: 802.11, Bluetooth and Handheld Devices. Recommendations of the National Institute of Standards and Technology; 2002.
2. Badra M, Serhrouchni A, and Urien P. A lightweight identity authentication protocol for wireless networks. *Computer Communications* 2004; vol. 27, issue 17, p. 1738-1745.
3. NFC Forum. Logical Link Control Protocol. Technical Specification, LLCP 1.1. 201.
4. Roland M. Software Card Emulation in NFC-enabled Mobile Phones: Great Advantage or Security Nightmare. *Fourth International Workshop on Security and Privacy in Spontaneous Interaction and Mobile Phone Use*; 2012. p. 1-6.
5. European Telecommunications Standards Institute. Smart Cards; UICC - Contactless Front-end (CLF) Interface; Part 1: Physical and data link layer characteristics. Release 9. TS 102 613 – V10.0.0; 2012.
6. ISO 7816. Cards Identification - Integrated Circuit Cards with Contacts; 2011.
7. Kfir Z and Wool A. Picking Virtual Pockets using Relay Attacks on Contactless Smartcard Systems. *SecureComm*; 2005. p. 47-58.
8. Smart Card Alliance. The Mobile Payments and NFC Landscape: A U.S. Perspective. Publication Number PC-11002; 2011.
9. Frisby W, Moench B, Recht B, and Ristenpart T. Security Analysis of Smartphone Point-of-Sale Systems. *USENIX Workshop on Offensive Technologies*. 2012; p. 22-33.
10. Dierks T and Rescorla E. Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246; 2008
11. Rosati T and G., Zaverucha G. Elliptic Curve Certificates and Signatures for NFC Signature Records. Research In Motion, Certicom Research; 2013.
12. Urien P. LLCPS: A new security framework based on TLS for NFC P2P applications in the Internet of Things. *IEEE Consumer Communications and Networking Conference*; 2013. p. 845-846.
13. Ahamad SS, Udgata SK, and Nair M. A Secure Lightweight and Scalable Mobile Payment Framework. International Conference on Frontiers of Intelligent Computing: Theory and Applications; 2013. p. 545-553.
14. Badra M and Hajjeh I. Key-exchange authentication using shared secrets. *IEEE Computer*; 2006. vol. 39, Issue 3, p. 58-66.
15. Vidas T, Votipka D, and Christin N. All Your Droid Are Belong To Us: A Survey of Current Android Attacks. USENIX Workshop on Offensive Technologies; 2011. p. 81-90.
16. Gummeson J, Priyantha B, Ganesan D, Thrasher D, and Zhang P. EnGarde: Protecting the Mobile Phone from Malicious NFC Interactions. 11th International Conference on Mobile Systems, Applications, and Services; 2013. p. 445-458.
17. Rieback M, et al. A Platform for RFID Security and Privacy Administration. USENIX Systems Administration Conference (LISA); 2006. p. 89-102.
18. Tamrakar S, Ekberg J.E and Asokan N. Identity Verification Schemes for Public Transport Ticketing with NFC Phones. STC; 2011. p. 37-48.
19. D. Abdul-Elminaam, H. Abdul-Kader and M. Hadhoud. Performance Evaluation of Symmetric Encryption Algorithms. *Communications of the IBIMA*; 2009. Vol. 8, ISSN: 1943-7765.
20. Badra, M. and I. Hajjeh. ECDHE_PSK Cipher Suites for Transport Layer Security (TLS); 2009. RFC 5489.
21. Badra, M. Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode. RFC 5487, DOI 10.17487/RFC5487, March 2009, http://www.rfc-editor.org/info/rfc5487.
22. Eronen, P. and H. Tschofenig. Pre-Shared Key Ciphersuites for Transport Layer Security (TLS). RFC 4279; December 2005.